

TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Marko Täht

Venni diagrammide õpiprogramm

Bakalaureusetöö (9 EAP)

Juhendaja:
dots. Rein Prank

Tartu 2016

Venni diagrammide õpiprogramm

Lühikokkuvõte:

Selles bakalaureusetöös luuakse õpiprogramm Venni diagrammide ülesannete lahendamiseks ja kontrollimiseks. Õpiprogrammiga saab lahendada kuute ülesandetüüpi, programm kontrollib lahendamisel tehtud vigu ja oskab lahendamisel ka abi pakkuda.

Võtmesõnad:

Venni diagramm, õpiprogramm

CERCS: S281 Arvuti õpiprogrammide kasutamise metoodika ja pedagoogika, P110 Matemaatiline loogika, hulgateooria, kombinatoorika

Educational software for Venn diagram

Abstract:

In this bachelor's thesis a learning program is created for Venn diagrams exercise solving and checking. Six exercise types are included where program can check four types of errors and can offer help for solving the exercise.

Keywords:

Venn diagram, learning program

CERCS: S281 Computer-assisted education, P110 Mathematical logic, set theory, combinatorics

Sisukord

1.	Sissejuhatus	4
2.	Probleemi taust	5
2.1	Õppeaine	5
2.2	Venni diagrammid	5
2.3	Hulgateooria põhitehted Venni diagrammidega	6
2.4	Probleemid Venni diagrammidega	8
3.	Sarnased programmid	10
3.1	Marten Hennochi bakalaureuse töö	10
3.2	Kolme hulga Venni diagrammi generaator	11
4.	Venni diagrammide ülesannete programmi kirjeldus	12
4.1	Ülesannete tüübid	12
4.2	Tudengi programm	12
4.2.1	Venni diagrammi koostamise dialoog	14
4.2.2	Avaldise leidmise dialoog	17
4.2.3	Tulemuste vaatamine	18
4.3	Õpetaja programm	21
4.3.1	Ülesandekogu loomine	21
5.	Programmi implementatsioon	24
5.1	Avaldis	24
5.2	Diagramm	24
5.3	Avaldise generaator	25
5.4	Automaatlahendaja	26
6.	Kokkuvõte	28
7.	Summary	29
8.	Kasutatud materjalid	30
	LISA I Keelefailide lisamine	31
	Lisa II Kokkupakitud fail näidisülesandekoguga, õpetaja programmiga ja õpilase programmiga	32
	Litsents	33

1. Sissejuhatus

Aines “Matemaatiline maailmapilt” õpetatakse erinevaid diskreetse matemaatika ja matemaatilise analüüsi osasid. Lausearvutuse õppimiseks on kasutuses mitu õpiprogrammi, mis muudavad mugavamaks ja kiirendavad harjutuste tegemist. Õpiprogrammide abil antakse tudengile erinevaid ülesandeid lausearvutuses. Ülesannete lahendamisel saab tudeng kohest tagasisidet. Lõplikest tulemustest saab teha ülevaate sellest, kui hästi on ülesandeid lahendatud.

Hetkel pole kasutuses programme hulgateooria ülesannete jaoks. Sellel aastal tehakse kaks bakalaureusetööd sellel teemal:

1. Diana Algma bakalaureuse töö, kus täiendatakse lausearvutuse ja predikaatarvutuse avaldiste teisendamiste programmi hulgateooria ülesannetega;
2. Käesolev bakalaureusetöö, kus luuakse programm Venni diagrammidega töötamiseks.

Bakalaureusetööna programmeeritud Venni diagrammide õpiprogrammis saab õppejõud luua ülesandekogusid eri ülesannetega. Programm võimaldab lahendada kuut ülesandetüüpi:

- 1) *avaldise leidmine Venni diagrammi järgi;*
- 2) *Venni diagrammi leidmine avaldise järgi;*
- 3) *hulkade võrdsuse kontroll;*
- 4) *alamhulga kontroll;*
- 5) *hulkade võrdsuse tingimuse leidmine;*
- 6) *alamhulga tingimuse leidmine.*

Lahendatud tudengifailide pealt saab õppejõud näha, milliseid ülesandeid lahendati hästi ja milliseid halvasti. Lisaks saab vaadata ka üksikute tudengite tulemusi ülesannete kogu raames.

Esimeses peatükis räägitakse, kuidas Venni diagramme kasutatakse ja kuidas need aitavad mõista hulgateooriat. Teises peatükis vaadeldakse olemasolevaid programme ja varem tehtud bakalaureusetöid samadel teemadel ning tuuakse välja nende puudused antud aine raames. Neljandas ja viiendas peatükis kirjeldatakse loodud programmi, kuidas erinevaid osasid realiseeriti ning miks neid nii loodi. Lisas on kaasa pandud programmi lähtekood.

2. Probleemi taust

2.1 Õppeaine

Tartu Ülikoolis õpetatakse ainet MTMM.00.342 Matemaatiline maailmapilt, kus õpetatakse esimese aasta informaatikutele, arvutitehnikutele ja matemaatikutele edaspidises õppes vajaminevaid teoreeme ja baasavaldisi. Aine on jaotatud kolmeks suureks teemaks:

- 1) lausearvutus,
- 2) hulgateooria,
- 3) funktsioonid.

Lausearvutuses õpitakse lausearvutuse põhitehteid ja avaldiste teisendamist ning nende tõeväärtustabeleid. Hulgateooria osas vaadeldakse hulgateooria põhitehteid, mõisteid ja Venni diagramme. Funktsioonide osas vaadeldakse, mis on funktsioonid, kuidas on funktsioonid seotud hulkadega ja mis on funktsioonide omadused. Aines toimuvad auditoorsed loengud ja praktikumid. Enamik praktikume on tahvlipraktikumid, kus lahendatakse ülesandeid käsi. Mõned praktikumid on arvutipraktikumid, kus töö toimub varem loodud programmi-dega. Hetkel on kasutuses kaks programmi: veebikeskkond tõeväärtustabeli ülesannete jaoks ja Java programm lausearvutuse teisendusülesannete jaoks. Kasutuses pole ühtegi programmi hulgateoreetiliste ülesannete jaoks. Käesolevas bakalaureusetöös luuakse programm Venni diagrammide ülesannete jaoks. Ülesandetüüpideks on:

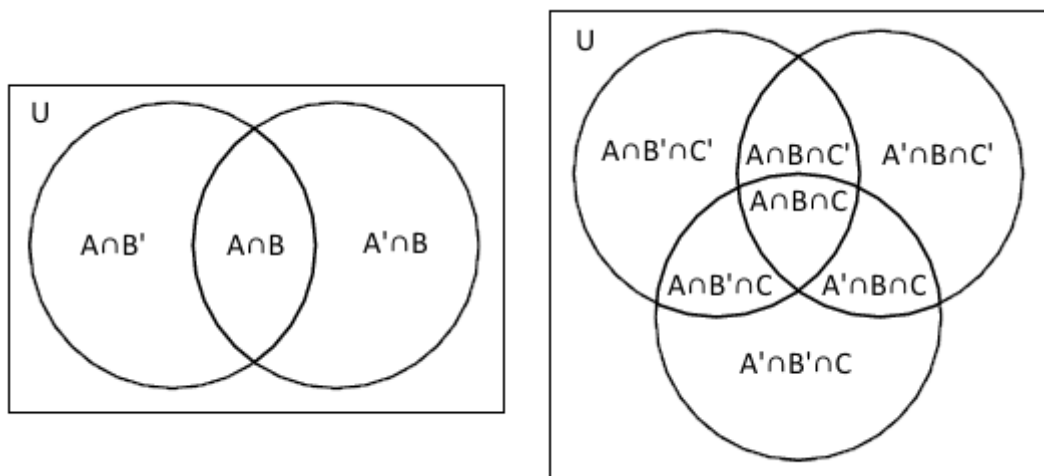
- 1) *Venni diagrammi leidmine avaldisele;*
- 2) *avaldise leidmine Venni diagrammist;*
- 3) *kahe hulga võrdsuse kontroll;*
- 4) *kahe hulga võrdsuse tingimuse leidmine;*
- 5) *alamhulga kontroll;*
- 6) *alamhulga tingimuse leidmine.*

Teisenduste programmi täiendab Diana Algma oma bakalaureusetöös, kus lisab juurde hulgateoreetilised ülesanded.

2.2 Venni diagrammid

Hulgateoreetiliste avaldiste graafiliseks kujutamiseks kasutatakse Venni diagramme, mis annavad kiire ja selge ülevaate hulgateoreetilistele avaldistele vastavatest hulkadest. Aines

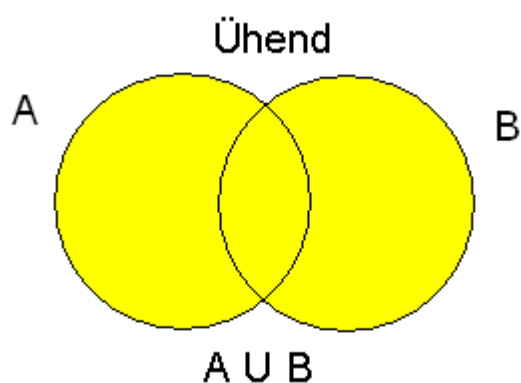
kasutatakse kahe ja kolme hulgaga diagramme. Hulki kujutatakse osaliselt kattuvate ringidena, kus kattuvad alad tähistavad elemente, mis kuuluvad mitmesse hulka. Kui avaldised sisaldavad ka täiendit, kujutatakse diagrammi ümber ka ristkülik (joonis 1), mis näitab universaalhuka, kuhu kuuluvad kõik hulgad.



Joonis 1. Kahe ja kolme hulgaga Venni diagrammid ja piirkondadele vastavad avaldised

2.3 Hulgateooria põhitehted Venni diagrammidega

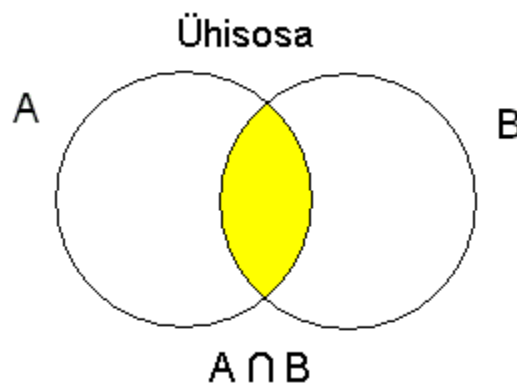
Programmis kujutatakse Venni diagramme ja nendega seotud hulgateoreetilisi tehteid. Käesolevas peatükis tuuakse välja hulgateooria põhitehted, milleks on täiend, ühisosa, vahe, ühend ja sümmeetriline vahe[1].



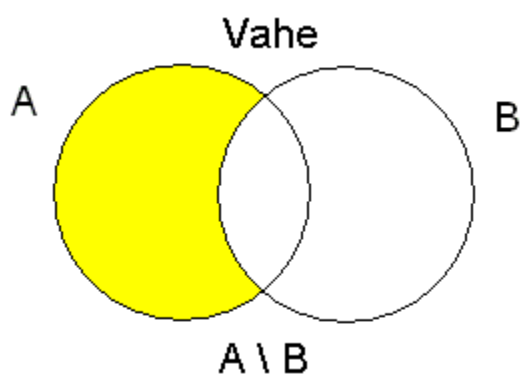
Def. Hulgade A ja B **ühendiks** ehk summaks nimetatakse hulka $A \cup B$, mille moodustavad kõik elemendid, mis kuuluvad vähemalt ühte hulgadest A ja B . Joonis 2.

Joonis 2. Hulgade A ja B ühend

Def. Hulkade A ja B **ühisosaks** ehk **lõikeks** nimetatakse hulka $A \cap B$, mille moodustavad kõik elemendid, mis kuuluvad nii hulka A kui hulka B . Joonis 3.

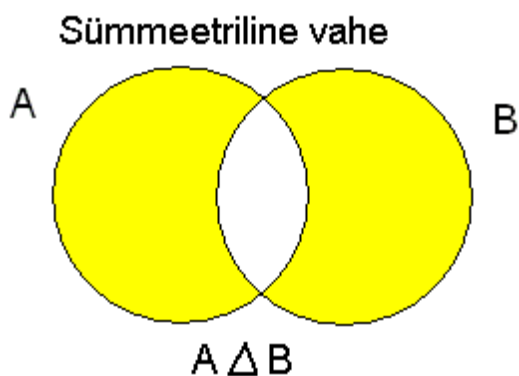


Joonis 3. Hulkade A ja B ühisosa



Joonis 4. Hulkade A ja B vahe

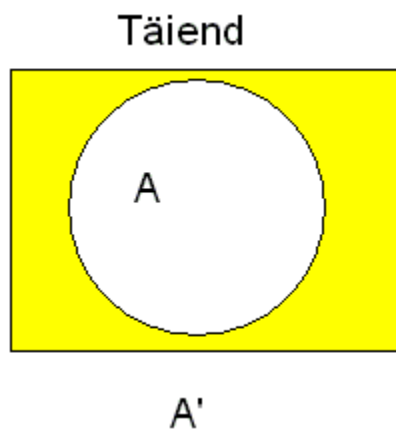
Def. Hulkade A ja B **vaheks** nimetatakse hulka $A \setminus B$, mille moodustavad kõik elemendid, mis kuuluvad hulka A , aga ei kuulu hulka B . Joonis 4.



Joonis 5. Hulkade A ja B sümmeetriline vahe

Def. Hulkade A ja B **sümmeetriliseks vaheks** nimetatakse hulka $A \Delta B$, mille moodustavad elemendid, mis kuuluvad parajasti ühte kahest hulgast A ja B . Joonis 5.

Def. Hulga A **täiendiks** A' (või A) nimetatakse hulka, mille moodustavad kõik need universaalse hulga elemendid, mis ei kuulu hulka A . Joonis 6.

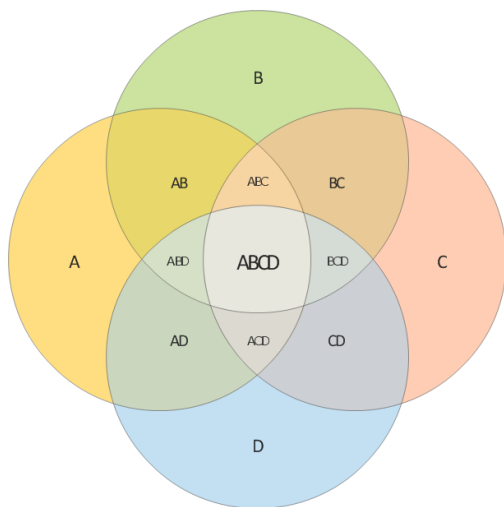


Joonis 6. Hulga A täiend

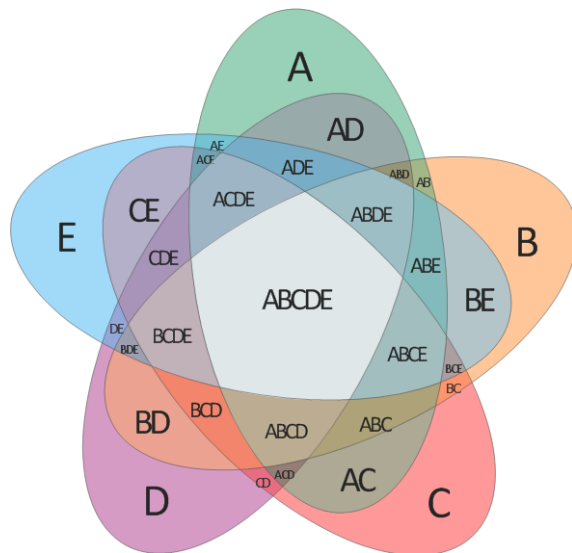
2.4 Probleemid Venni diagrammidega

Venni diagramm on efektiivne moodus saada kiiret ja selget ülevaadet avaldisest. Probleemiks võivad osutuda paljude tehetega avaldised, sest uute diagrammide joonistamine on aeganõudev. Rohkem kui kolme hulgaga ülesannete lahendamine Venni diagrammidel võib tekitada probleeme, kuna on vaja leida keerukaid kujusid, et tekiksid kõik ühisosade alad.

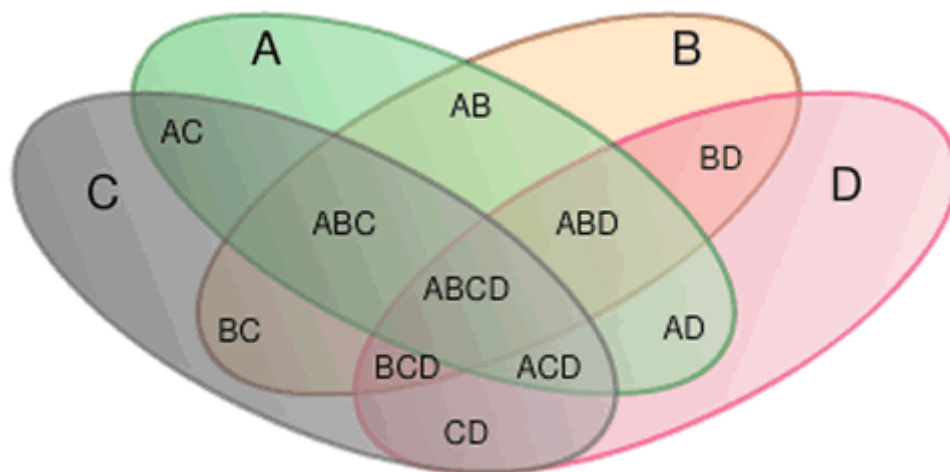
Joonistel kujutatud diagrammidel kasutatakse teistsugust tähistust kui Tartu Ülikoolis. Tähis AB nelja hulgaga diagrammil tähendab $A \cap B \cap C' \cap D'$. Nelja hulgaga diagramme on veel võimalik arusaadavalt kujutada (joonis 9 [4]), kuid kerge on teha neis vigu. Joonisel 7 [2] oleval diagrammil on puudu kaks osahulka: $A \cap B' \cap C \cap D'$ ja $A' \cap B \cap C' \cap D$. Viie hulgaga Venni diagrammide (joonis 8 [3]) puhul lähevad osad alamhulgad väga väikeseks ja nendega muutub opereerimine keerukaks. Kuna aine raames rohkem kui kolme hulgaga ülesandeid ei lahendata, siis ei muutu ülesanded ka liialt keeruliseks ja ülesannete lahendamine Venni diagrammidega on väga mõistlik.



Joonis 7. Nelja hulgaga vigane Venni diagramm



Joonis 8. Viie hulgaga Venni diagramm



Joonis 9. Korrektne nelja hulgaga Venni diagramm

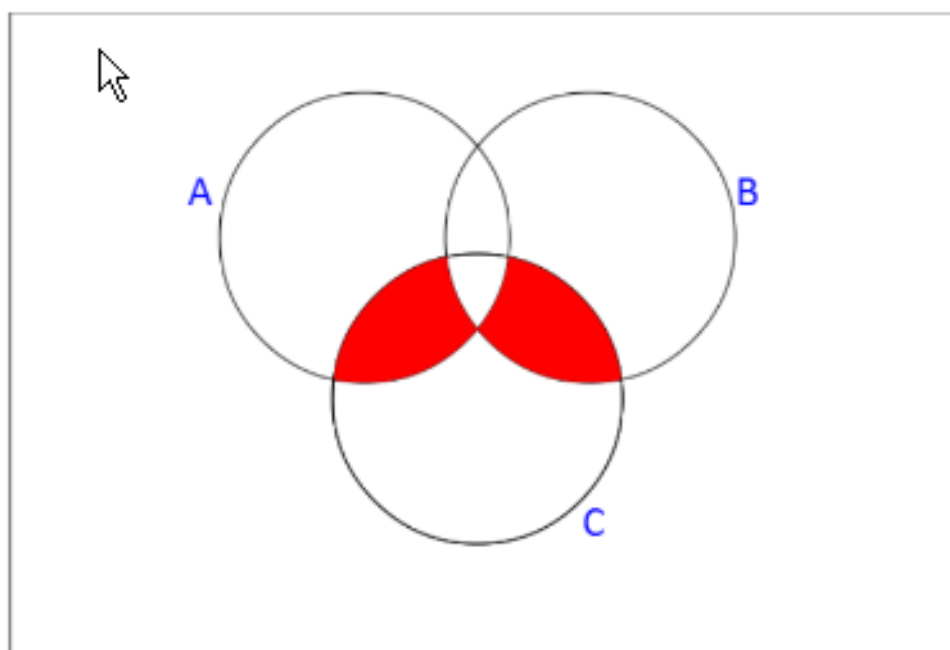
3. Sarnased programmid

3.1 Marten Hennochi bakalaureuse töö

2012. aastal tegi Marten Hennoch oma bakalaureusetöö [5] sarnasel teemal, kus ta lisas olemasolevale tõeväärtustabeli ülesannete veebikeskkonnale juurde Venni diagrammide ülesandetüübid. Lisati neli ülesandetüüpi:

- 1) *valemi leidmine diagrammi järgi,*
- 2) *diagrammi leidmine valemi järgi,*
- 3) *hulkade võrdsuse kontroll*
- 4) *kontrollida, kas üks hulk on teise hulga alamhulk.*

Sisestada antud diagrammile vastav avaldis



Tehtemärgid

Klahv:	1	2	3	4	5
Kopeeri:	\cup	\cap	Δ	\setminus	$'$

Vastus:

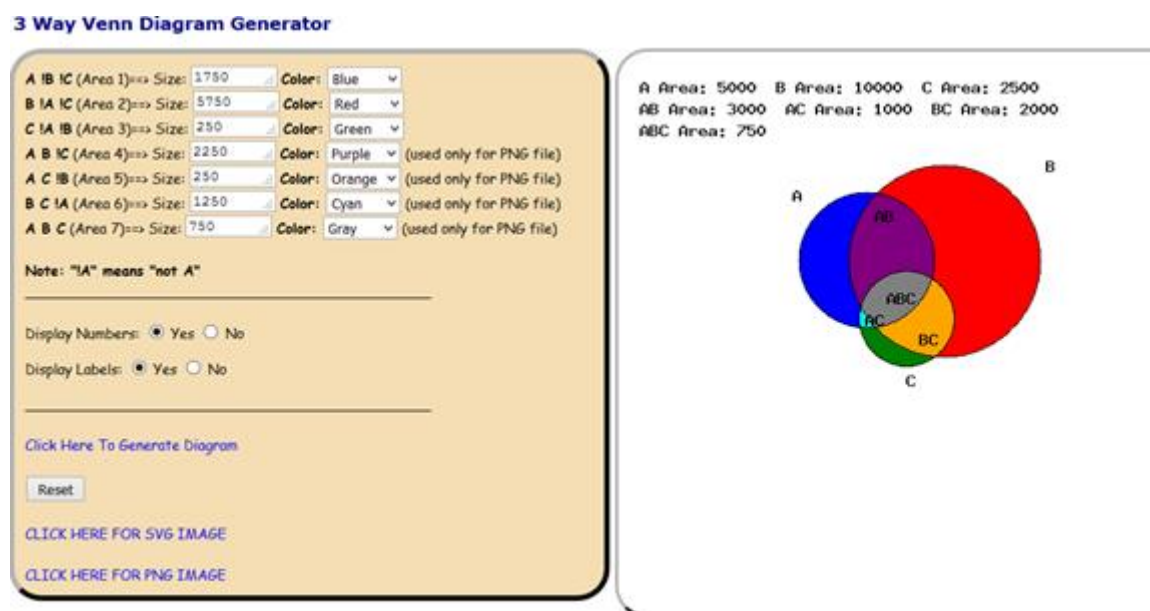
Joonis 10. Hennoch'i programmi ülesandetüüp *valemi leidmine diagrammi järgi* [5]

Hennoch'i loodud täienduse plussiks on, et tegu on veebikeskkonnaga, seega tudengid ei saa tulemuste faile teisiti mõjutada, kui ainult lahendades ülesandeid. Lisaks, kuna on tegu veebikeskkonnaga, siis pole vaja programmi oma arvutisse laadida ja pole vaja saata faile

juhendajale, et juhendaja saaks neid kontrollida. Kõik on kättesaadav internetist vastavate õigustega kasutajate poolt. Põhiliseks miinuseks on ülesandetüüpide vähesus ja vahetulemuste kujutamise võimaluste puudumine.

3.2 Kolme hulga Venni diagrammi generaator

MIT(Massachusetts Institute of Technology) Bioinformaatika leheküljel on veebipõhine Venni diagrammide generaator [6].Antud generaator genereerib diagrammi vastavalt sisestatud andmetele. Andmeid, mida programm soovib on osahulkade elementide arvud ja osahulga värvid, et neid oleks kergem eristada pildi peal. Tähistus $A \setminus B \setminus C$ tähistab hulka $A \cap B' \cap C'$.



Joonis 11. MIT Bioinformaatika Venni diagrammide generaator

Programmi heaks osaks on olemasolu veebilehel. Genereeritud Venni diagrammidel on iga hulk eri suurusega, mis näitab ära, kui suur on mingi hulk võrreldes mõne teise hulga.

Antud programm ei sobi Tartu Ülikooli ainesse, sest diagrammi loomiseks on vaja osahulkade elementide arvu, aga aines tegeletakse üldiselt abstraktsete hulkadega, seega ei ole neis hulkades kindlaid elemente ja puudub ka kindel suurus. Kui panna kõikide hulkade elementide arvuks sama arv, siis luuakse Venni diagramm, mida saaks kasutada näitena, aga ülesandeid nende peal ei saaks teha. Tegu on rohkem programmiga, mida kasutada uurimistöödel, et visualiseerida seoseid, kui programmiga, millega saaks õpetada Venni diagramme.

4. Venni diagrammide ülesannete programmi kirjeldus

4.1 Ülesannete tüübid

Loodud programmis on 6 ülesandetüüpi:

- 1) *Venni diagrammi leidmine avaldise järgi;*
- 2) *avaldis leidmine Venni diagrammi järgi;*
- 3) *hulkade võrdsuse kontroll;*
- 4) *hulkade võrdsuse tingimuse leidmine;*
- 5) *alamhulga kontroll;*
- 6) *alamhulga tingimuse leidmine.*

Ülesandetüübid 1 ja 2 on baasülesandetüübid.

Ülesandetüübid 3,4,5,6 on ülesandetüübi 1 laiendused.

Ülesandetüübil 1 põhineb lahendus Venni diagrammi koostamisel etteantud avaldise järgi, tüübil 2 valemi koostamisel Venni diagrammi järgi.

Venni diagrammi konstrueerimist nõudvatel tüüpidel konstrueeritakse diagramm sammude kaupa. Selleks on olemas kaks diagrammi leidmise režiimi:

- 1) ühe tehte kaupa diagrammi leidmine;
- 2) vabalt alamvalemi valikuga diagrammi leidmine.

Ühe tehte kaupa diagrammi leidmisel tuleb igal sammul realiseerida Venni diagrammina üks avaldises olev tehe. Joonisel 12 on diagrammi konstrueerimisel tehtud kolm sammu ja neljandal sammul on esimesele diagrammile komponendid korrektselt märgitud ja teisele diagrammile on alamavaldisele vastav diagramm märgitud.

Vaba tehte valikuga saab valida alamvalemi ja sellele koostada diagrammi. Alamavaldisel valiku maksimaalne suurus on 3 sellist osahulka, millel on juba eelnevalt loodud diagramm või mis on üksik hulga muutuja, sest kui osahulki on rohkem, siis võivad diagrammil olevad märgistused liiga kirjuks minna ja ei saaks enam eriti hästi aru, kus miski on.

4.2 Tudengi programm

Tudengi programmis saab lahendada ülesandeid ja vaadata tudengifailis ühe ülesandekogu kohta olevaid tulemusi ja lahendusi.

Venni diagrammi koostamise ülesannetes koosneb lahendussamm neljast osast:

- 1) osaavaldise valikust osaavaldise valiku lahtris;
- 2) osaavaldise komponentide märkimisest diagrammile uue diagrammi loomise alas;
- 3) osaavaldisele vastava diagrammi koostamine uue diagrammi loomise alas;
- 4) koostatud diagrammi esitamine vajutades nupule „Valmis“.

Sammu igal osal toimub vigade kontroll. Osaavaldise valikul kontrollitakse, kas valitud osaavaldis on süntaktiliselt korrektne ja kas on järgitud tehete järjekorda. Komponentide märkimisel diagrammile, osaavaldisele vastava diagrammi koostamisel ja koostatud diagrammi esitamisel kontrollitakse, kas diagrammid märgistati õigesti.

Tehete järjekorra viga tekib, kui valitakse avaldise osa, mis ei ole avaldise alamavaldis.

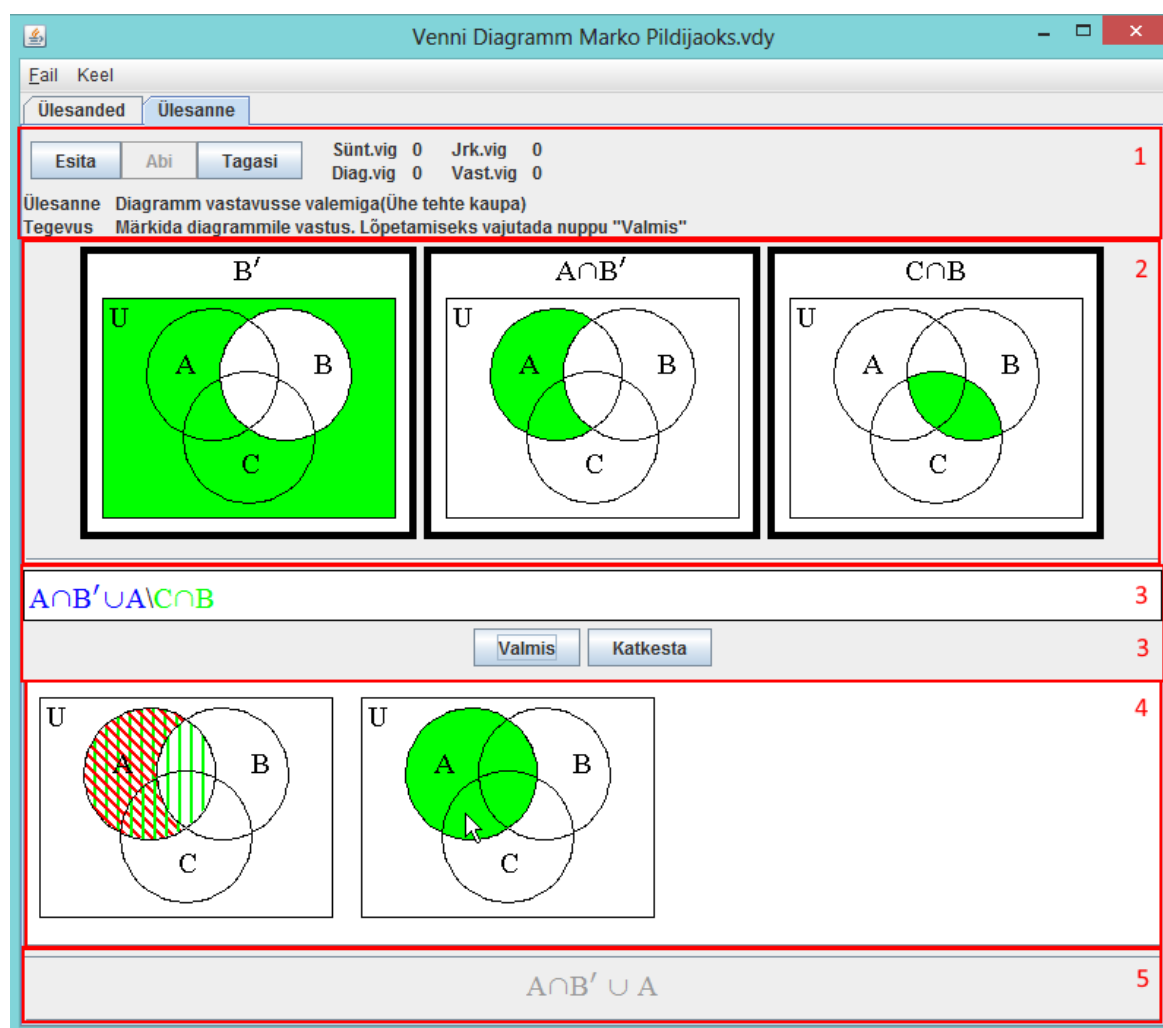
Samm loetakse lõppenuks, kui esitatud diagramm vastab osaavaldisele. Konstrueeritud diagrammid paigutatakse valminud diagrammide alale valmimise järjekorras. Osaavaldise valikuväljas värvitakse diagrammiga osavaldis roheliseks. Diagrammi konstrueerimiseks valitud osaavaldis on värvitud siniseks. Lõppvastuseks loetakse viimasena konstrueeritud diagrammi. Lõppvastuse esitamisel kontrollitakse, kas koostatud diagramm vastab avaldisele.

Diagrammile avaldise leidmise ülesannetes koosneb samm avaldise esitamisest. Esitamisel kontrollitakse, kas esitatud avaldis on süntaktiliselt korrektne ja kas esitatud avaldis vastab diagrammile.

Baasülesandetüüpide järgi on ülesannete lahendamiseks loodud kaks dialoogi:

- 1) Venni diagrammi koostamise dialoog;
- 2) avaldise leidmise dialoog.

4.2.1 Venni diagrammi koostamise dialoog



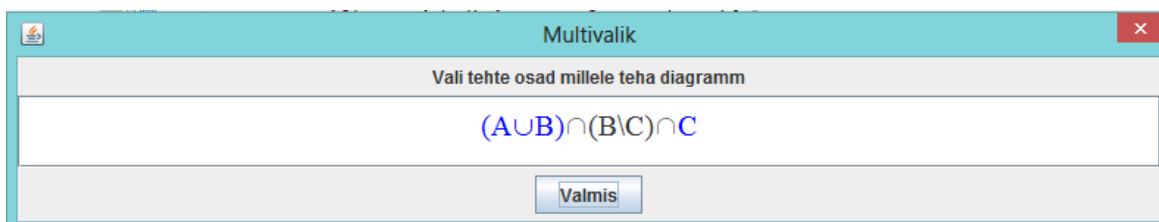
Joonis 12. Venni diagrammi koostamise dialoog

Venni diagrammi koostamise dialoogi aken (joonis 12) on jagatud 5 osaks.

Esimene osa koosneb ülesande menüüst, infost vigade kohta ja juhistest. Ülesande menüü koosneb kolmest nupust: „Esita“, „Abi“ ja „Tagasi“. Nupuga „Esita“ saab esitada lõppvastust. Nupp „Abi“ annab nõu, kuidas tuleks ülesannet lahendada või millist sammu järgmisena teha. Nupp „Abi“ on aktiivne ainult siis, kui ülesandekogu looja on selleks ülesande loomisel loa andnud. Nupp „Tagasi“ võtab tagasi viimase tehtud sammu. Nupp „Tagasi“ töötab ainult siis, kui ei ole käimas mõnele alamavaldisele diagrammi konstrueerimist. Lahenduse infost on näha tehtud vigade arv, ülesande sisu ja alamavaldise valiku režiim ning juhised ülesande lahendamise etapi jaoks.

Teine ülesande lahendamise akna osa on valminud diagrammide ala. Kui mõnele alamavaldisse luuakse korrektne Venni diagramm, lisatakse loodud diagramm siia alasse. Diagrammid on järjestatud loomise järjekorras. Et vältida korduvatele alamavaldistele diagrammide konstrueerimist, saab siin märgistada diagrammi ja seejärel osaavaldisel valikut tehes teostatakse kontroll, kus otsitakse, kas märgistatud diagrammile vastav avaldis leidub valitud osaavaldises. Kui valitud avaldis leidub ja see pole osa mõnest pikemast avaldisest, millel on diagramm olemas, siis pannakse diagramm vastavusse selle avaldisega.

Kolmandas alas on osaavaldisel valiku lahter. Siin saab valida osaavaldist, millele soovitakse luua diagrammi. Osaavaldised, millele on loodud diagramm, on värvitud roheliseks. Siniseks on värvitud osaavaldis, millele hetkel diagrammi konstrueeritakse. Kui on tehtud osaavaldisel valik, siis peale klahvi „Enter“ või nupu „Vali“ vajutamist lukustub avaldisel lahter ja nuppude „Multivalik“ ja „Vali“ asemele tekivad kaks uut nuppu „Valmis“ ja „Katkesta“. Nupp „Valmis“ kontrollib, kas diagrammile on hulga õigesti märgistatud ja seejärel ilmub tühi diagramm, kuhu saab kanda alamavaldisel vastavat diagrammi. Vajutades teist korda nupule „Valmis“ kontrollitakse, kas vastuse diagramm vastab alamavaldisel, ja seejärel esitatakse tehtud diagramm. Nupp „Katkesta“ katkestab hetkel valitud alamavaldisel diagrammi loomise või juhul, kui on mindud vastamise etappi, läheb tagasi märgistamise etappi. Sarnaselt nupule „Vali“ ja klahvile „Enter“ töötab nupp „Multivalik“. Nupp „Multivalik“ võimaldab teha ühendis või ühisosas tehteid teises järjekorras kui vasakult paremale. Joonisel 13 on näha, et valitud on osad $(A \cup B)$ ja C on värvitud siniseks. Kui sobivad osad on valitud tuleb vajutada „Valmis“. Sellepeale organiseeritakse valemiosad ümber, et valitud osad oleksid avaldises järjest. Edasi toimub tegevus nagu tavalise valiku puhul



Joonis 13. Multivaliku aken

Neljas osa on uue diagrammi loomise ala. Diagrammi loomine matki tööd paberil, kus kõigepealt märgitakse ühel joonisel erinevate tähistuste/värvide abil avaldisel komponendid ja seejärel märgitakse teisel joonisel avaldisel vastavad ühisosas. Siin alas, kui on valitud osaavaldis, tekib tühi Venni diagramm. Diagrammile tuleb märkida osaavaldisel osad, selleks tuleb avaldises valida osa, mida märkida, ja vajutada vastavatele aladele diagrammil.

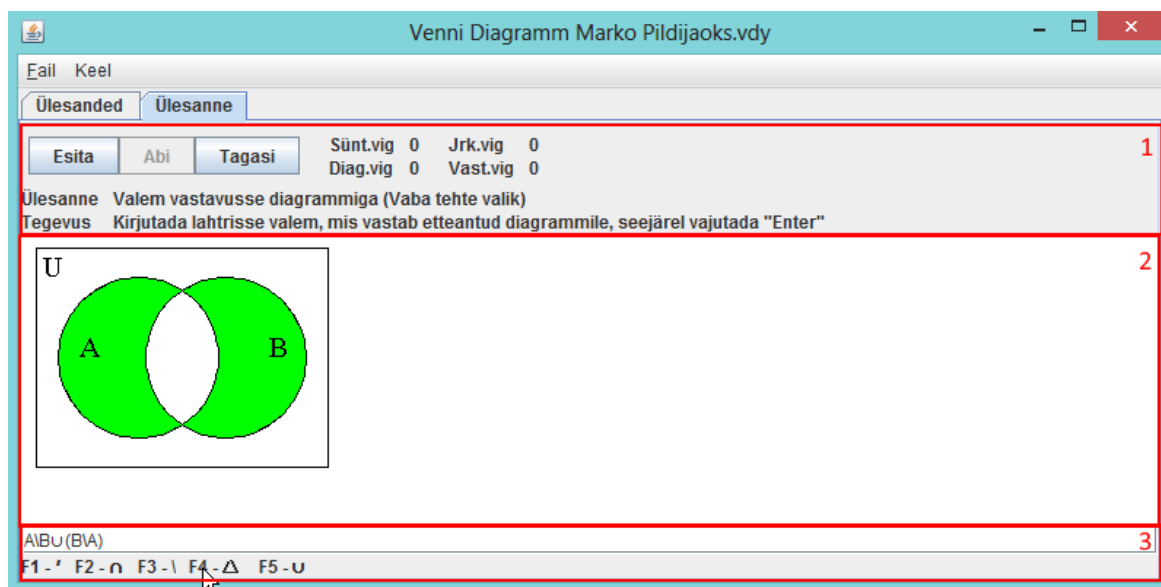
Kui diagrammile on märgitud valitud osaavaldise osad, tekib nupule „Valmis“ vajutades teine tühi Venni diagramm esimese kõrvale. Teisele diagrammile tuleb märkida valitud osaavaldisele vastavad ühisosad. Joonisel 12 on osaavaldise osad märgitud esimesele diagrammile ja teisele diagrammile on märgitud tulemusele vastavad ühisosad.

Viies osa on osaavaldise osade valiku ala. Kui kolmandas osas on valitud osaavaldis, töödeldakse osaavaldis läbi ja leitakse kõik lahendatud osad või üksikud avaldise muutujad. Seejärel kuvatakse osaavaldis osadena. Halliks värvitud osad on tehtemärgid ja sulud, mis ei kuulu osavalemite juurde, neid ei saa valida diagrammile märkimiseks. Mustaks värvitud osad on diagrammidega alamhulgad või üksikud avaldise muutujad. Valides mustaks värvitud osa, muudab see värvi. Seda värvi, mis on valitud osa, tehakse ka triipe diagrammile. Et diagrammile teha osade märgistust, peab olema siit valitud mõni osaavaldise osa. Kui esimesele diagrammile on edukalt märgitud kõik avaldise osad, siis osade valik lukustub. Vastuse märkimiseks tuleb lihtsalt vajutada diagrammi alade peale. Valikus saab korraga olla kuni kolme osaga alamavaldis, sest igat alamavaldist tähistatakse diagrammile erinevalt ja kui võimalik oleks valida rohkem kui kolm, läheks diagramm väga kirjuks ja diagramm ei oleks enam nii ülevaatlik.

Hulkade võrdsuse kontroll, alamhulga kontroll, hulkade võrdsuse tingimuse leidmine, alamhulga tingimuse leidmine

Kontrolli ja tingimuse leidmise ülesanded kasutavad Venni diagrammi koostamise dialoogi, et esmalt luua diagrammid avaldise mõlemale poolele. Nupule „Esita“ vajutamisel esitatakse jah/ei küsimus. Võrdusega ülesannete puhul küsitakse, kas hulgad on võrdsed ja alamhulga ülesannetes küsitakse, kas üks hulk on teise alamhulk. Vastates kontrollitakse, kas antud vastus on õige. Kontrolli ülesande puhul õige vastuse korral loetakse ülesanne lõppenuks. Tingimuse leidmise ülesannete puhul õige vastuse korral, kui vastuseks oli „Ei“, peab tudeng näitama, millised alamhulgad peavad olema tühjad, et kehtiks võrdus või üks hulk oleks teise alamhulk.

4.2.2 Avaldise leidmise dialoog



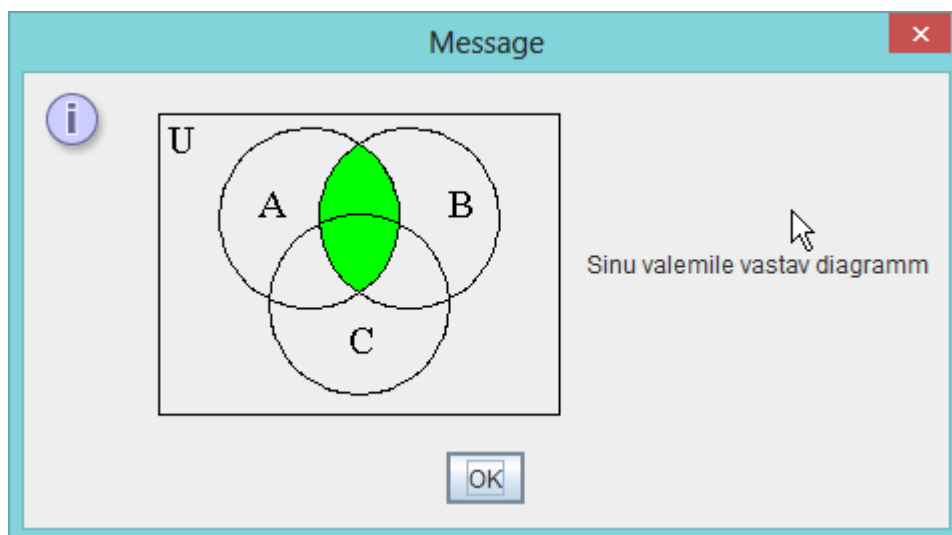
Joonis 14. Avaldise leidmise dialoog

Avaldise leidmise dialoogi aken (joonis 14) on jagatud kolmeks osaks.

Esimene osa on sama, mis ülesande aknas, kus leitakse avaldise järgi Venni diagramm. Ainsaks erinevuseks on see, et abi nuppu ei saa kasutada, sest Venni diagrammile avaldise leidmise ülesandes pole abi andmist sellisel kujul.

Teine osa on diagrammi ala, kus näidatakse diagrammi, millele tuleb luua avaldis.

Kolmas osa on avaldise sisestusala. Sisestamisel lubatakse ainult sümboleid A, B, C, (,) ja hulgateoreetilisi tehtemärke. Vajutades klahvi „Enter“ või nuppu „Esita“, kontrollitakse, kas sisestatud avaldis on süntaktiliselt korrektne ja kas avaldis kehtib näidatud Venni diagrammile. Õigeks loetakse kõik avaldised, mis vastavad näidatud diagrammile. Kui sisestatud valem ei vasta näidatud diagrammile, näidatakse tudengile, millist diagrammi tema valem kirjeldab (joonis 15).



Joonis 15. Teade, kus näidatakse, millisele diagrammile vastab tudengi sisestatud avaldis

4.2.3 Tulemuste vaatamine

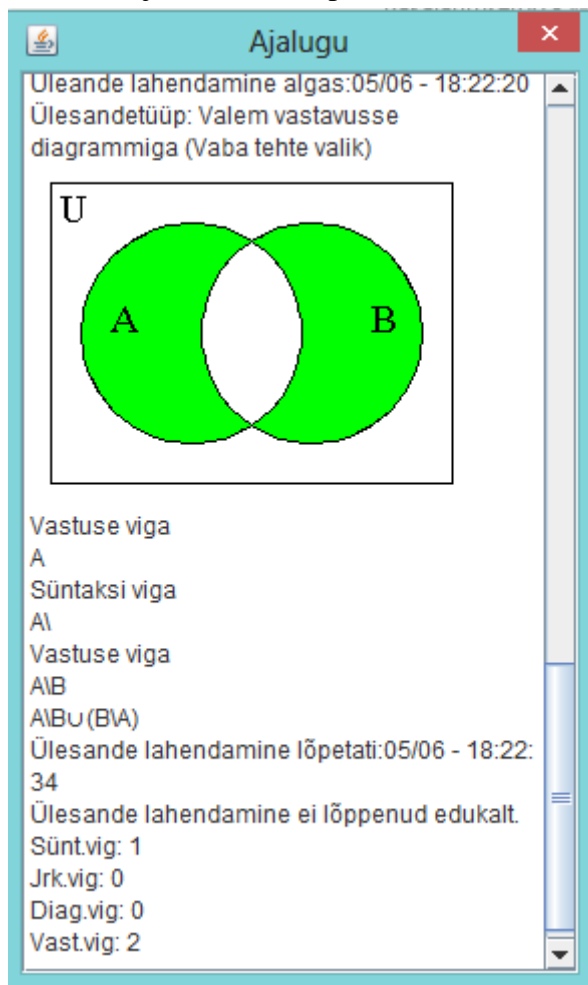
Tudeng saab vaadata oma ülesannete kogu ülesannete tulemuste tabelit (joonis 16), kus ta näeb iga ülesande lahenduse kohta oma erinevate vigade arvu, katsete arvu, mitu korda on ülesande lahendamist katkestanud ja kas ülesande on loetud lahendatuks või mitte.

Nr.	Lahen...	Katseid	Katke...	Sünt.vig	Jrk.vig	Vast.vig	Diag.v...
1	Valmis	1	0	0	1	0	0
2	Valmis	2	1	1	1	0	2
3	Ei ole ...	1	0	0	0	2	0
4	Ei ole ...	0	0	0	0	0	0
5	Ei ole ...	0	0	0	0	0	0
6	Ei ole ...	0	0	0	0	0	0
7	Ei ole ...	0	0	0	0	0	0
8	Ei ole ...	0	0	0	0	0	0
9	Ei ole ...	0	0	0	0	0	0

Joonis 16. Ülesannete kogu tulemuste tabel

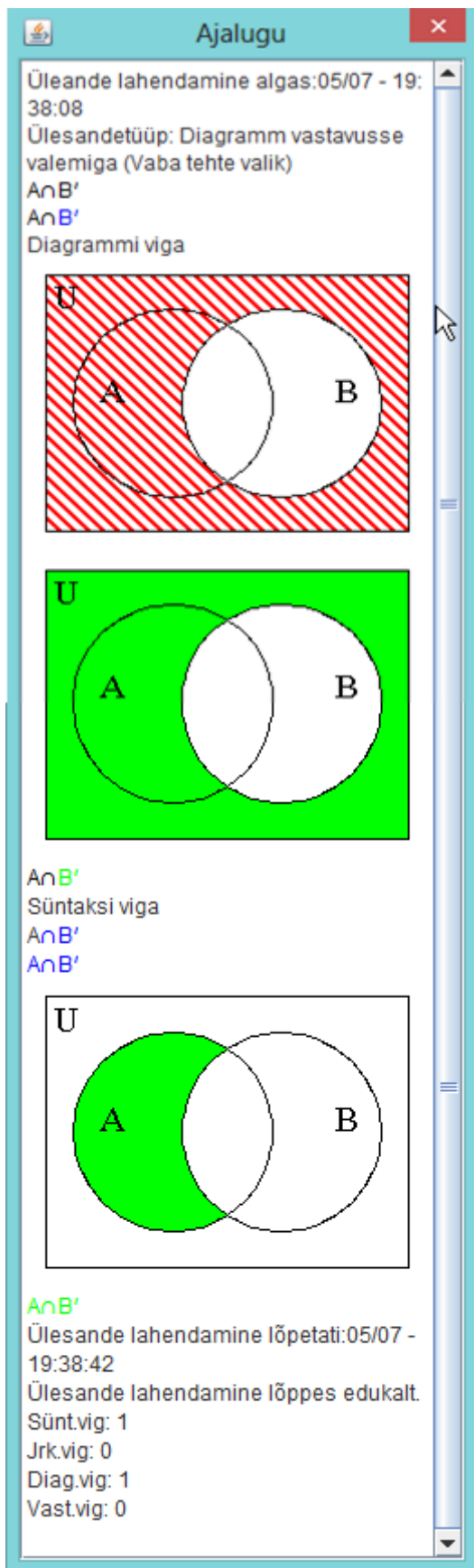
Topeltklõps rea peale avab ülesande lahenduskäigu ajaloo. Ajaloost on näha samme, mida tudeng on teinud, et ülesannet lahendada.

Ajaloo esimesel real on ülesande lahendamise alguse kellaaeg ja kuupäev. Teisel real on toodud välja ülesandetüüp.



Joonis 17.1 Venni diagrammile avaldise leidmise ülesande lahenduse ajaloo aken

Venni diagrammile avaldise leidmise ülesande lahenduskäigu aknas (joonis 17.1) kolmandal real on näidatud diagramm, millele hakatakse avaldist otsima. Järgnevatel ridadel on tudengi poolt tehtud pakkumised. Kui tudeng on teinud süntaktilise vea, on kiri „Süntaksi viga“ ja järgmisel real on vigane avaldis. Kui tegu oli süntaktiliselt korrektse avaldisega, aga avaldis ei kirjeldanud näidatud diagrammi, on kiri „Vastuse viga“ ja järgneval real on sisestatud avaldis. Kui sisestatud avaldis oli süntaktiliselt korrektne ja kirjeldas ka näidatud diagrammi, on näidatud ainult avaldis.



Diagrammile avaldise leidmise ülesande lahenduse ajaloo akna (joonis 17.2) kolmandal real on avaldis, millele hakatakse diagrammi konstrueerima. Järgmisel real on tudengipoolt tehtud osaavaldise valik. Kui tehti viga diagrammi osade märgistamisel, siis näidatakse vigast diagrammi, kus alad on triibutatud. Kui viga tehti osaavaldisele diagrammi konstrueerimisel näidatakse vigast diagrammi. Kui loodi korrektne diagramm osaavaldisele, näidatakse korrektset diagrammi ja selle alla avaldist, kus on näidatud roheliseks värvitud ala, millele konstrueeritud diagramm vastab. Kui osaavaldise valikul tehti viga, siis öeldakse vea tüüp ja näidatakse tehtud osaavaldise valikut. Peale viimast diagrammi näidatakse kogu avalist rohelisena.


Lahendus lõpus on ajaloo aknal ülesande lõpetamise kellaaeg ja kuupäev. Peale mida öeldakse, kas ülesande lahendamine lõppes edukalt või mitte. Järgmistel ridadel on tehtud vigade arvud. Kui ülesannet lahendati rohkem kui ükskord järgneb vigade arvu loendile järgmise lahendamise lahenduskäik.

Joonis 17.2 Diagrammi leidmise ülesande lahenduse ajaloo aken

4.3 Õpetaja programm

Õpetaja programm võimaldab luua uusi ülesandekogusid, millele saab lisada ülesandeid, ja vaadata tudengite tulemusi ülesandekogude lahendamise kohta. Ülesannetele on võimalik genereerida avaldisi ja diagramme. Tudengi vastuste põhjal on võimalik vaadata tulemuste üldtabelit. Täpsemaks vaatluseks on võimalik ka näha tudengi lahenduskäike koos tehtud vigadega.

4.3.1 Ülesandekogu loomine

Vajutades nuppu „Uus ülesanne“ , avaneb uus aken (joonis 18), milles saab määrata ülesande erinevaid parameetreid:

- 1) ülesandetüüp;
- 2) diagrammi koostamise tüüp;
- 3) harjutuse tüüp;
- 4) lubatud vigade arvud;
- 5) automaatlahendaja abipakkumise lubamine/keelamine;
- 6) ülesande tegevus;
- 7) ülesande sisu vastavalt ülesandetüübile:
 - a. kui ülesandetüüp on avaldisele diagrammi leidmine, saab anda ette avaldise (joonis 18) või seadistada, kuidas avaldist genereerida;
 - b. kui ülesandetüüp on diagrammile avaldise leidmine, saab märkida erinevaid diagrammi osi või seadistada, kui mitu tehtemärki on diagrammi lühimas valemis (joonis 19);

- 8) kui ülesande parameetrid on määratud, siis nupu „Valmis“ vajutamisel luuakse uus ülesanne ja paigutatakse ülesandekogusse.

Uus ülesanne

Ülesandetüüp

☒ Valem diagrammiks ☐ Diagramm valemiks

☐ Võrdsuse kontroll ☐ Alamhulga kontroll

☐ Võrdsuse tingimuse leidmine ☐ Alamhulga tingimuse leidmine

Diagrammi koostamise tüüp

☒ Ühe tehte kaupa ☐ Vaba osaavaldise valik

Harjutuse tüüp

☐ Kontrollitöö ☒ Harjutamine

Max vigade arv

Sünt.vig	5	Kaal	30
Jrk.vig	5	Kaal	40
Diag.vig	5	Kaal	20
Vast.vig	5	Kaal	50

Abi

☐ Paku ☒ Ei paku

Tegevus

Diagramm vastavusse valemiga

Ülesande sisu

Valem

☒ Sisestatav ☐ Genereeritav

A∩B'∪A∩C

F1 - ' F2 - ∩ F3 - \ F4 - Δ F5 - ∪

Valmis Katkesta

Joonis 18. Ülesande loomise aken avaldisele Venni diagrammi konstrueerimise ülesande jaoks

Joonis 19. Ülesande loomise aken Venni diagrammile avaldise leidmise ülesande jaoks

4.3.1.1 Tulemuste vaatamine

Tulemuste vaatamisel on kaks vaadet:

- 1) ühe tudengi ülesanded ühe ülesandekogu lõikes, nagu tudengi programmis;
- 2) kõikide tudengite lahendused antud ülesandekogu lõikes.

Kõikide tudengite lahenduste vaates on üldtabel (joonis 20). Üldtabeli nägemiseks tuleb valida kaust, kus on tudengi failid ja valida ülesannete kogu, mille kohta soovitakse näha tudengite tulemusi. Üldtabelis on kõikide tudengite nimed, rühmad ja ülesande lahendamise edukuse hinnang vahemikus 0-1. Hinnang arvutatakse tehtud vigade pealt. Igale veale saab määrata erineva kaalu, mis mõjutab hinnangut erinevalt. Vajutades ülesande hinnangu peale, avaneb lahenduskäigu vaade (joonised 17.1 ja 17.2).

nimi	Rühm	1	2	3	Summa
Marko	1	1.0	0.75	1.0	2.75

Joonis 20. Tudengite tulemuste üldtabel

5. Programmi implementatsioon

Programm loodi kasutades programmeerimiskeelt Java ja sellesse sisseehitatud graafiliste liideste teeki Swing. Java versiooniks on Java 1.7, juhuks, kui mõnel kasutajal ei ole viimast Javat või on mõni muu probleem uuemate Java versioonidega.

Projektil on kaks peameetodit. Üks käivitab tudengi programmi ja teine käivitab õppejõu programmi. Mõlemad programmid jagavad klasse, mis defineerivad ülesandeid, ülesandekogusid ja diagrammi. Õpetaja programmi spetsiifiliseks osaks on ülesandekogude loomine ja ühe ülesandekogu raames kõikide tudengite tulemuste vaatamine. Tudengi programmi spetsiifiliseks osaks on ülesannete lahendamine ja ühe tudengi tulemuste vaatamine ühe ülesandekogu raames.

Järgnevad peatükid kirjeldavad tudengiprogrammi põhilisi klasse.

5.1 Avaldis

Kuna avaldist ainult sõne kujul hoida ei saa, sest lahendamise ajal saadud vahepealseid diagramme oleks raske seostada kindlate avaldiste osadega ja avaldise puuna hoidmine on mäluulukas ja raskendab mitmeid operatsioone, on loodud avaldise jaoks eraldi klass.

Klassis hoitakse avaldise sõne kujul ja avaldisega seotud Venni diagramme hoitakse Tree-Map struktuuris, kus iga Venni diagrammi jaoks on võti kahe väärtusega. Võtme väärtused määravad avaldises alamavaldisi lõiku, mille kohta valitud Venni diagramm kehtib.

Kui ülesande lahendamisel avaldise väljas valitakse alamavaldis, siis tekstiväljas tehtud valiku algus- ja lõppindeksid saadetakse avaldise klassi. Avaldise klassis teostatakse kontroll, kas valitud alamavaldis on korrektne alamavaldis.

Kui alamavaldis on korrektne, tagastatakse kõik alamavaldisse kuuluvate tehtud tehete diagrammid õiges järjekorras.

5.2 Diagramm

Diagrammi määrab neli klassi. Esimene klass on Hulk, mida kasutatakse esialgse tühja Venni diagrammi joonistamiseks. Teine klass on Ala, millega lõigatakse joonistatud diagramm osadeks. Igale osale vastab ainult üks alamhulk kogu diagrammist. Kolmas klass on Stiili klass, kus toimub diagrammi osade värvimine ja triibutamine. Neljas klass on Diagrammi klass, mis ühendab eelmist kolme klassi. Diagrammi klassi on salvestatud osaavaldis,

mille kohta diagramm käib. Diagrammi klass oskab ise arvutada avaldisele vastavat diagrammi ning oskab leida diagrammile lühimat valemit.

Valemile diagrammi leidmiseks parsitakse avaldis avaldisepuuks. Avaldisepuu läbitakse lõppjärjestuses. Lõppjärjestuses läbimine tähendab, et enne kontrollitakse kõik alluvad ja seejärel tipp ise.

Diagrammile vastava lühima avaldise leidmiseks pannakse diagramm vastavusse binaarse numbriga. Käsitsi on määratud kolme baashulga binaarid: A, B ja C. Nende kolme binaariga hakatakse tegema loogilisi tehteid, kuni jõutakse binaarini, mis vastab diagrammi binaariga. Ühisosa jaoks tehakse disjunktsioon, ühendi jaoks konjunktsioon ja täiendi jaoks eitus. Näiteks, kui on kolme hulga Venni diagramm ja diagrammil on märgitud hulk A, oleks antud diagrammi binaarseks numbriks 11110000. Kolme hulga Venni diagrammi binaarne number koosneb kaheksast numbrist, sest diagrammil on kaheksa erinevat ala. Iga number tähistab ühte kindlat ala ja vastavalt, kas number on 1 või 0, ala kuulub diagrammi või mitte. Numbrid tähistavad järgmist:

- 1) $A \cap B' \cap C'$
- 2) $A \cap B \cap C'$
- 3) $A \cap B \cap C$
- 4) $A \cap B' \cap C$
- 5) $A' \cap B \cap C'$
- 6) $A' \cap B \cap C$
- 7) $A' \cap B' \cap C$
- 8) $A' \cap B' \cap C'$

5.3 Avaldise generaator

Ülesande sisu	
Valem	
<input type="radio"/> Sisestatav <input checked="" type="radio"/> Genereeritav	
Hulkade arv	3
Ühendeid	3
Ühisosasid	3
Vaheid	1
Sümeetrilisi vaheid	2
Täiendeid	1

Joonis 21. Avaldise generaatori sätete aken

Ülesande loomise aknas, avaldise juhuslikuks genereerimiseks tuleb valida „Genereeritav“ ja seejärel seadistada generaator. Generaatori sätetest luuakse vastav kuuekohaline arv, mis edastatakse generaatorile.

Generaatoris on määratud 7 reeglit, mille järgi genereeritakse avaldis. Reegliteks on:

- 1) $\rightarrow F$
- 2) $F \rightarrow F'$
- 3) $F \rightarrow (F)'$
- 4) $F \rightarrow F \setminus F$
- 5) $F \rightarrow F \cap F$
- 6) $F \rightarrow F \Delta F$
- 7) $F \rightarrow F \cup F$

Avaldise genereerimist alustatakse alati esimesest reeglist. Vastavalt kuuekohalisele numbrile, leitakse tehted, mida lisada. Igale tehtele vastab üks või kaks reeglit. Kui on leitud reeglid, mida saab rakendada, valitakse juhuslikult nende seast üks reegel ja F asendatakse valitud reeglga. Kui valitud reegel pole täiendi või ühisosa moodustamine, lisatakse 50% tõenäosusega tehtele ümber sulud. Kui reegel on rakendatud, vähendatakse võimalike tehete arvust selle tehte arvu, et oleks teada, mitu tehet on veel vaja lisada. Protsess kestab, kuni pole enam ühtegi tehet lisada.

Seejärel asendatakse kõik F -id tähtedega A, B või A, B, C sõltuvalt sellest, kas on tegu kahe või kolme hulga avaldisega. Asendused toimuvad nii, et poleks tehteid, mille kaheks argumentiks on üks ja sama täht. Peale genereerimist eemaldatakse avaldisest üleliigsed sulud. Näiteks, kui valemis on $(A)'$, pole siin sulge vaja ja need eemaldatakse.

5.4 Automaatlahendaja

Automaatlahendaja ülesandeks on leida avaldisest järgmised alamavaldised, millele võiks luua diagrammi, et jõuda lähemale lõpptulemusele. Järgmise sammu otsing toimub nelja sammuna.

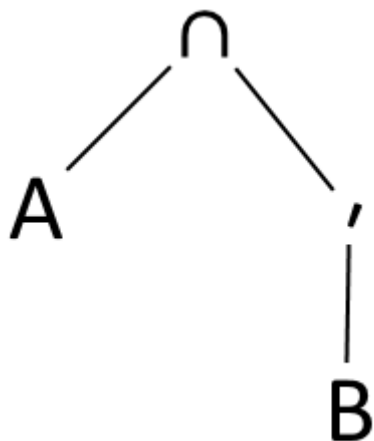
Esimene samm on avaldise parsimine avaldisepuuks. Avaldisepuuks on kahendpuu, mille keskjärjestuses läbimisel saab tagasi esialgse avaldise.

Teises etapis lisatakse avaldise puusse diagrammid neile alampuudele, millel on diagrammid olemas. Seda on kerge teha, sest diagrammid on organiseeritud neile vastavate alamavaldiste algus- ja lõppindeksi järgi, ning puudes igasse tippu on salvestatud tema indeks avaldises. Seega on vaja leida tipp, mis jääb diagrammi avaldise algus- ja lõppindeksite vahele ja kinnitada tipule diagramm.

Kolmandas etapis läbitakse puu eesjärjestuses. Iga tipu juures esmalt kontrollitakse, kas tema kõigil alluvatel on olemas diagramm või on tegu ühe tähega. Kui ei ole, minnakse esmalt vasakusse alluvasse, seejärel paremasse alluvasse ja teostatakse sama kontroll.

Kui leitakse sobiv tipp, küsitakse vahemikku, mida see tipp valemis esindab ja jäetakse meelde. Otsitakse kõik võimalikud järgmised sammud.

Viimase sammuna tagastatakse nimekiri indeksitest, mille vahel leiduvad alamavaldistes on sobilikud järgmise sammu tegemiseks.



Joonis 22. Avaldise $A \cap B'$ avaldisepuu.

6. Kokkuvõte

Bakalaureusetöö eesmärgiks oli luua programm Venni diagrammi ülesannete loomiseks, lahendamiseks ja kontrollimiseks. Töö raames loodi programm, millel on kuus ülesandetüüpi:

- 1) *avaldis leidmine Venni diagrammile;*
- 2) *Venni diagrammi leidmine avaldisele;*
- 3) *kahe avaldis võrdsuse kontroll Venni diagrammidega;*
- 4) *kahe avaldis võrdsuse tingimuse leidmine Venni diagrammidega;*
- 5) *kahe avaldis alamhulga kontroll Venni diagrammidega;*
- 6) *kahe avaldis alamhulga tingimuse leidmine Venni diagrammidega.*

Igat ülesandetüüpi, peale *avaldis leidmine Venni diagrammile*, saab lahendada kahes alamvalemi valiku režiimis: ühe tehete kaup režiim ja vaba osaavaldis valiku režiim.

Iga lahendusammu juures kontrollitakse avaldise ja diagramme. Avastatud vea korral nõutakse, et see parandataks. Programm loendab nelja tüüpi vigu: süntaktiline viga, tehete järjekorra viga, diagrammi viga, vastamise viga.

Kõik tegevused ülesande lahendamisel salvestatakse tudengifaili, kust hiljem saab õppejõud vaadata tudengi tulemust ja lahenduskäiku.

Lahendamisel pakub abi automaatlahendaja, mis soovitab samme, mida võiks teha.

Diagrammide jaoks oskab programm otsida lühimat avaldist, mis vastab diagrammile.

Programmi loomisel prooviti leida häid lahendusi erinevate osade loomiseks. Disaini aluseks võeti juba olemasolev lausearvutuse teisenduste programm, et uue programmi õppimine oleks kiirem ja lihtsam ning välimus oleks juba tuttav.

7. Summary

MTMM.00.342 Transition to Advanced Mathematics is a course taught in Tartu University to first year computer science and mathematics students. As a part of this course set theory is taught. Some of the practices are done with computers, where learning programs are used. Currently there are no programs for set theory in use.

The aim of this bachelor's thesis is to create a program, in which students can learn about Venn diagrams. The program has six exercise types:

- 1) *creating formula for Venn diagram;*
- 2) *creating Venn diagram for formula;*
- 3) *checking if two sets are equal using Venn diagrams;*
- 4) *finding condition when two sets are equal using Venn diagrams;*
- 5) *checking if one set is a subset to another set using Venn diagrams;*
- 6) *finding condition when one set is a subset to another set using Venn diagrams.*

All of the exercise types, with the exception of *Creating formula for Venn diagram*, have two subformula selection types: free subformula selection and one operation at a time selection.

On each step of solving the exercise error check is done. If error is found it is wanted that the error to be fixed. In total there are four error types: syntax error, order of operations error, diagram error and answering error. All errors made are counted.

All the actions student makes while solving the exercise are saved in the result file, where teacher can see how the student solved the exercise.

While solving the exercise an automatic solver can offer help on what step to take next.

For diagrams program can find the shortest formula that corresponds to the diagram.

The application was created to visually resemble existing Algebraic Manipulation Assistant for Propositional Logic and Predicate Calculus program, to make it easier to learn and more familiar to use.

8. Kasutatud materjalid

- [1] Hõim, T. Hulga mõiste. TÜ aine Matemaatiline Maailmapilt konspekt, Moodle, 2015
- [2] 4-Set Venn diagram – Template <https://conceptdraw.com/a389c3/preview> (6.05.2016)
- [3] 5-Set Venn diagram - Template <https://conceptdraw.com/a390c3/preview--5-Set%20Venn%20diagram%20template> (6.05.2016)
- [4] How to Deal with Four Sets in Set Theory <http://www.mbatious.com/prepnote/how-deal-four-sets-set-theory> (6.05.2016)
- [5] Marten Hennoch, Venni diagrammidega seotud ülesannete tüübid diskreetse matemaatika veebikeskkonna jaoks. Tartu Ülikool 2012.
<http://dspace.ut.ee/bitstream/handle/10062/32909/thesis.pdf> (9.04.2016)
- [6] 3 Way Venn Diagram Generator <http://jura.wi.mit.edu/bioc/tools/venn3way/> (6.05.2016)

LISA I Keelefailide lisamine

Programm on kokku pakitud *.jar* formaati, mida saab käivitada. Käesolev juhend kehtib ekstraheerimistarkvarade kohta, mis toetavad *drag-n-drop* failide ekstraheerimist ja juurdelisamist. Teiste ekstraheerimistarkvarade puhul võib juhend veidi erineda. Keelefaili lisamine on testitud programmi *WinRAR* peal. Et lisada või teha koopiaid keelefailides, saab *.jar* faile avada tavalise ekstraheerimise tarkvaraga. Faili ennast mitte ekstraheerida, vaid lihtsalt avada, et näha, mis on seal sees. Failis on kaust „res“, kus on keelefailid. Keelefaili saab kätte lihtsa *drag-n-drop* moodusega. Avades keelefaili, on näha tekst kujul:

New = New

File = File

Exit = Exit

Save = Save

Load = Load

openexset = Open exercise list

moveup = Move exercise up

movedown = Move exercise down

New_Exercise = New Exercise

edit = Edit

delexercise = Delete Exercise

newexset = New Exercise List

...

Kokku on keelefailis 137 rida. Keelefaili tõlkimiseks tuleb asendada peale märki „=“ olevad tekstid vastava tõlkega soovitud keelde. Keelefaili nimi peab olema kujul *Labels_en_EN*, kus en ja EN on asendatud tõlgitud keele tähisega. Keelefaili paigaldamiseks tagasi faili tuleb uus keelefail *drag-n-drop* moodusega lohistada *.jar* faili „res“ kausta.

Uus keelefail on valmis kasutamiseks.

**Lisa II Kokkupakitud fail näidisülesandekoguga, õpetaja
programmiga ja õpilase programmiga**

Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina, **Marko Täht**,

(autori nimi)

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose
Venni diagrammide õpiprogramm,
(lõputöö pealkiri)

mille juhendaja on Rein Prank,

(juhendaja nimi)

- 1.1.reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
- 1.2.üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace'i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartus, **12.05.2016**